Robotics Research Technical Report

Generatorium omnis laboris ex machina

Coordinated Motion Planning for
Two Independent Robots

by

Micha Sharir
Shmuel Sifrony

Technical Report No. 360
Robotics Report No. 148
March, 1988

NYU COMPSCI TR-360 c.1
Sharir, Micha
Coordinated motion planning
for two independent robots

New York University
Courant Institute of Mathematical Sciences

Computer Science Division
251 Mercer Street New York, N.Y. 10012

Coordinated Motion Planning for
Two Independent Robots

by

Micha Sharir
Shmuel Sifrony

Technical Report No. 360
Robotics Report No. 148
March, 1988

New York University
Dept. of Computer Science
Courant Institute of Mathematical Sciences
251 Mercer Street
New York, New York 10012

Work on this paper has been supported by a research grant from the Joint Ramot — Israeli Ministry of Industry Foundation, and by a research grant from the NCRD — the Israeli National Council for Research and Development. Work on this paper by the first author has also been supported by Office of Naval Research Grant N00014-87-K-0129, by National Science Foundation Grant No. NSF-DCR-83-20085, by grants from the Digital Equipment Corporation, and the IBM Corporation.

Coordinated Motion Planning for Two Independent Robots

Micha Sharir^(1,2) and Shmuel Sifrony⁽²⁾

⁽¹⁾ Courant Institute of Mathematical Sciences, New York University

⁽²⁾ Computer Science Department, Tel-Aviv University

ABSTRACT

We present an $O(n^2)$ algorithm for planning a coordinated collision-free motion of two independent robot systems of certain kinds, each having two degrees of freedom, which move in the plane amidst polygonal obstacles having a total of n corners. We exemplify our technique in the case of two "planar Stanford arms", but also discuss the case of two discs or convex translating objects. The algorithm improves previous algorithms for this kind of problems, and can be extended to a fairly simple general technique for obtaining efficient coordinated motion planning algorithms.

1. Introduction

In this paper we present a new approach to the design of efficient algorithms for coordinated collision-free motion planning for two independent robot systems moving amidst a collection of obstacles which they must avoid (and also avoid collision into one another). We exemplify our techniques in the case of two "planar Stanford arms", where each arm is modeled as a line segment which can slide forward and backward through some fixed point, and can also rotate about that point; thus each arm has two degrees of freedom (see Fig. 1). The arms are assumed to be moving in the plane amidst a collection of polygonal obstacles, having a total of n corners. This problem has been studied by Fortune, Wilfong and Yap [FWY], who present an $O(n^3)$ algorithm for it. We present here an improved algorithm whose complexity is only $O(n^2)$. This algorithm is a special case of a more general technique that we introduce here, and which can be applied to obtain efficient algorithms for various other coordinated motion planning problems. For example, we obtain an $O(n^2)$ algorithm for coordinating the motion of two independent discs (or more generally of two translating convex objects) in the plane, amidst polygonal obstacles as above, thereby matching and extending a previous result of Yap [Ya1], as a special case of our more general approach.

Work on this paper has been supported by a research grant from the Joint Ramot - Israeli Ministry of Industry Foundation, and by a research grant from the NCRD - the Israeli National Council for Research and Development. Work on this paper by the first author has also been supported by Office of Naval Research Grant N00014-87-K-0129, by National Science Foundation Grant No. NSF-DCR-83-20085, by grants from the Digital Equipment Corporation, and the IBM Corporation,

Another recent work on coordinated motion planning is [EL]; they use a more heuristic approach, whose worst case performance is inferior to that of our technique. We also obtain nearly quadratic algorithms for coordinated motion planning for practically *any* pair of independently moving systems with two degrees of freedom each, and several other efficient coordinated motion planning algorithms for pairs of more complex systems, or even of larger groups of such systems.

As in many standard motion planning algorithms, our goal is to obtain a discrete combinatorial representation of the space FP of all free placements of our system, and decompose that space into its (arcwise) connected components, so that, given an initial and final placements of the system, we can check whether they both lie in the same connected component of FP (which is exactly the condition for a continuous collision-free motion to exist between these two placements), and if so we can also plan such a motion. See [SS1], [SS2], [SS3], [OSY1], [OSY2], [LS], [KS], [SiS], [Ya1], [FWY], [KO] for a sample of recent algorithms of this kind.

One of the standard approaches, initially proposed in [SS1], [SS2], and used in many recent papers, is to obtain a *cell decomposition* of FP into simple and connected cells, determine adjacency of pairs of these cells, and represent this structure by a *connectivity graph*, whose vertices are these cells, and whose edges connect pairs of adjacent cells. Then any instance of motion planning between a specified pair of an initial and final placements of our system can be easily reduced to a purely combinatorial path-searching problem in the connectivity graph.

We can exemplify the cell decomposition approach in the case of coordinated motion planning for two independent systems B_1, B_2 , each having two degrees of freedom, and both moving in a 2-D polygonal space whose boundary has n corners. This will motivate our approach, and will also serve to introduce some terminology. We note that FP is four-dimensional, and we decompose it by first calculating $FP(B_1)$, which is the (two-dimensional) space of all free placements of B_1 , ignoring the presence of B_2 . For each placement $X \in FP(B_1)$, let us denote by $FP_X(B_2)$ the two-dimensional "fiber" space of all free placements of B_2 , when B_1 is fixed at placement X (i.e. it is regarded as another fixed obstacle). Now if we vary X continuously within $FP(B_1)$, the combinatorial representation of $FP_X(B_2)$ will remain the same, unless X crosses one of finitely many *critical curves*, at which the structure of $FP_X(B_2)$ changes discontinuously. In the case where B_1 and B_2 are two planar Stanford arms, it is shown in [FWY] that the number of critical curves is $O(n)$, that they are all algebraic of some fixed (and small) maximal degree, and that they thus partition $FP(B_1)$ into $O(n^2)$ *non-critical regions*, namely the connected components (i.e. faces) of the arrangement of these critical curves. It is further shown in [FWY] that the combinatorial complexity of each fiber space $FP_X(B_2)$ is $O(n)$, so that, in particular, it consists of $O(n)$ connected components $C_1(X), C_2(X), \dots$, with the property that the combinatorial representation of each component remains the same as X varies within a non-critical region. Thus, for each non-critical region R of $FP(B_1)$, we obtain $O(n)$ simple connected cells within FP , all of which project onto R ,

and each having the form $\{(X,Y) \mid X \in R, Y \in C_j(X)\}$ for some j . The collection of all these cells, over all non-critical regions R , yields the desired cell decomposition of FP . The same approach has also been used in [SS3] to obtain an algorithm for planning the coordinated motion of two independent discs.

Unfortunately, the total combinatorial complexity of all the resulting cells is $O(n^3)$ (both for the case of two Stanford arms and for the case of two discs). That this is far from being optimal has been demonstrated by Yap [Ya1] for the case of two discs, by giving an $O(n^2)$ (specially-tailored) algorithm for this case.

In this paper we introduce a new, and surprisingly simple, approach to cell decomposition of FP for two independently moving objects, and use it to obtain an improved $O(n^2)$ algorithm for the case of two Stanford arms. As a matter of fact, our technique is fairly general and can be applied to many coordinated motion planning problems involving two independently moving systems. For example, for the general case of two independent systems, each with two degrees of freedom, moving in a 2-D polygonal space with n corners (assuming that the complexity of each moving system is small and fixed), we obtain a nearly quadratic algorithm (see below for a more precise formulation). Efficient algorithms for various other special types of system are discussed in the concluding section 6.

2. Overview of our coordinated motion planning technique

Here is a description of our approach. Suppose B_1, B_2 are two systems with two degrees of freedom each. We first calculate $FP(B_1)$ and $FP(B_2)$ (defined in an analogous manner), and decompose each of these spaces into a collection of (pairwise openly disjoint) simple connected cells, such that each cell is bounded by a constant number of arcs of small and fixed maximal algebraic degree; let $\{c_1^1, \dots, c_{n_1}^1\}$ and $\{c_1^2, \dots, c_{n_2}^2\}$ be the resulting decompositions for $FP(B_1)$ and $FP(B_2)$ respectively. (Each of these calculations is a motion planning problem for a system having only two degrees of freedom, and can be solved efficiently either by a general procedure presented in [GSS] (and reviewed below), or by other efficient ad-hoc techniques; see below for additional discussion of this issue.) We also assume that the number of adjacent pairs of cells $(c_i^1, c_{i'}^1)$ (resp. $(c_j^2, c_{j'}^2)$) is only $O(n_1)$ (resp. $O(n_2)$). This is a natural assumption for 2-D problems, and follows generally from Euler's formula.

Next, for each $1 \leq i \leq n_1$ and $1 \leq j \leq n_2$, let FP_{ij} denote the intersection of FP with the Cartesian product $c_i^1 \times c_j^2$. Since both c_i^1 and c_j^2 are bounded by $O(1)$ curves, each of which describes either a contact of B_1 or of B_2 with some fixed obstacle feature, or else is an artificial cell boundary, we argue that FP_{ij} can be regarded as the free configuration space of the combined system B_1, B_2 , when moving in the presence of only $O(1)$ obstacles, or, more precisely, subject to only $O(1)$ collision-constraints. Thus, assuming that B_1 and B_2 have both $O(1)$ combinatorial complexity, it follows that the combinatorial complexity of each FP_{ij} is constant, and that a decomposition of it into

$O(1)$ 4-D cells (e.g. of the form mentioned above), together with determination of the adjacency of these cells (within FP_{ij}) can be obtained in constant time. (At worst such a decomposition can be obtained using general (but inefficient) techniques such as in [SS2], [Ca]; alternatively, more efficient ad-hoc decomposition techniques are preferred; they will improve the constant size of each FP_{ij} , but will not change the overall asymptotic complexity of the algorithm.)

Next, we "glue" all these pieces of FP together; this amounts to determining which cells within one such piece FP_{ij} are adjacent to which cells in some other "neighboring" piece $FP_{i'j'}$. This is also a fairly simple task, which proceeds as follows. We argue that for a cell Q within some FP_{ij} to be adjacent to a cell Q' within another $FP_{i'j'}$, it is necessary that either $i=i'$ and $c_{ij}^1, c_{i'j'}^1$ are adjacent in $FP(B_2)$, or that $j=j'$ and $c_{ij}^2, c_{i'j'}^2$ are adjacent in $FP(B_1)$. Thus there are only $O(n_1 n_2)$ pairs of pieces of FP that have to be glued, and the gluing of each such pair can be accomplished in constant time, since it involves a pair of pieces, each having $O(1)$ many cells of a constant total complexity (so that we can tell in $O(1)$ time which pairs of these cells are adjacent).

We thus obtain a decomposition of FP into $O(n_1 n_2)$ simple connected cells, such that the number of adjacent pairs of these cells in FP is also $O(n_1 n_2)$. This enables us to represent FP by a connectivity graph CG (as above) whose size is $O(n_1 n_2)$, and then to use CG to reduce any motion planning instance to path searching through that graph. This completes our general solution of the coordinated motion planning problem at hand. We remark that our technique is closely related to a technique recently proposed in [SS4] concerning motion planning in environments with "bounded local complexity".

2.1. Applications to special cases and some technical details

The quantities n_1, n_2 defined above are both $O(n)$ for the case of Stanford arms, and for the case of discs, or, more generally, for the case of a pair of convex polygons (bounded by a constant number of edges) translating independently in the plane. For the case of Stanford arms, this was shown in [FWY] and in [SiS]. The claim for discs or for translating convex polygons follows from the results of [KLPS]. Moreover, in both cases one can calculate the spaces $FP(B_1), FP(B_2)$ in subquadratic (actually close to linear) time, which is thus dominated by the quadratic cost of constructing the "product" space FP . We thus obtain $O(n^2)$ algorithms for all these problems.

For general systems with two degrees of freedom, n_1 and n_2 might be quadratic in n , if we calculate the entire 2-D spaces $FP(B_1), FP(B_2)$. However, extending a basic observation in [PSS] and [GSS], there is no need to calculate these spaces in their entirety; it suffices to obtain the connected component C_i of $FP(B_i)$ which contains the initial placement Z_i of B_i , for $i=1,2$, because any free placement of, say B_1 , which is reachable from Z_1 by a continuous collision-free motion (whether with B_2 present or not) must lie in C_1 , and similarly for B_2 . Thus it suffices to apply the cell decomposition

method described above to the two components C_1, C_2 rather than to the entire spaces $FP(B_1), FP(B_2)$. But, as shown in [GSS], the combinatorial complexity of a single component C of the free configuration space of a system B having two degrees of freedom, is $O(\lambda_s(n))$, where n is the number of geometric constraints on the free placements of B (all assumed to be representable by algebraic curves of low and constant maximal degree), s is a fixed parameter depending on the degree of these constraints (more precisely, it is $2 +$ the maximum number of intersections between any pair of these curves), and $\lambda_s(n)$ is the maximum length of (n, s) Davenport-Schinzel sequences, which is nearly linear in n for any fixed s (see [HS], [ASS] for more detail). Moreover, such a component can be calculated (under an appropriate model of computation) in time $O(\lambda_s(n) \log^2 n)$ [GSS]. Hence, applying these results to our case, we conclude that both n_1 and n_2 can be taken to be nearly linear in n , and that the relevant portion of FP within $C_1 \times C_2$ (which includes all placements of the combined system reachable from (Z_1, Z_2)) has only $O(\lambda_{s_1}(n_1)\lambda_{s_2}(n_2))$ combinatorial complexity (for appropriate parameters s_1, s_2), and can be calculated within a similar amount of time (which also dominates the time needed for the initial calculation of C_1, C_2).

In the following sections of the paper we provide details about the calculation of each FP_{ij} . Even though this takes only constant time per "piece", one should aim to make this time as small as possible, or else the constant of proportionality in the time bound for the algorithm may be unacceptably high. For those who really scorn any such consideration, we can recommend the very general Collins' cell decomposition technique, as used in [SS2], which runs in time doubly-exponential in the number of degrees of freedom (which is nevertheless constant in our case), or the improved general procedure of Canny [Ca], which has only singly-exponential time complexity. However, we will illustrate in the following sections more explicit and efficient decomposition techniques for the case of two Stanford arms and for the case of two discs.

The remainder of this paper is organized as follows. In section 3 we analyze the free configuration space of a single planar Stanford arm, as a preparatory step to the motion coordination algorithm, which is presented in section 4. Section 5 discusses the case of two independent discs (or translating convex objects), and section 6 contains concluding remarks, including various extensions of our techniques. Some technical details about the algebraic structure of critical curves arising in the case of two Stanford arms are given in Appendix A.

3. The free configuration space of one planar Stanford arm

We now exemplify the general technique outlined above in the special case of two independent planar Stanford arms. For this we first analyze the case of a single arm. A single planar Stanford arm A is modeled as a line segment PQ which is constrained to touch a fixed point C , and thus has two degrees of freedom - rotation about C and translation along its length

(through C). The workspace D of A is the disc centered at C and having radius $|PQ|$, and the circle surrounding D is denoted by E . D is assumed to contain polygonal obstacles, each of which is bounded by a closed simple polygonal curve composed of straight *walls* (edges) and of *corners* (vertices), such that any pair of obstacles has disjoint interiors, and the total number of their corners is n . Each placement of A at which P does not lie at C (including non-free placements) can be uniquely represented by the placement in D of the tip P of A . For simplicity of exposition (but with no real loss of generality), we exclude placements at which P coincides with C , so $D - \{C\}$ is a representation of the space of placements of A . (Our analysis can be easily adapted to cover cases in which there are some physical limits on the possible motions of A , e.g. when it can slide only up to certain length or rotate only within some restricted range of orientations, etc.) The free configuration space $FP(A)$ of A contains placements at which A does not touch any obstacle or (again for exposition sake) any point of the circle E . Let $BFP(A)$ be the boundary of $FP(A)$. It is shown in [FWY] (and also derived in Lemma 4.1 of [SiS]) that the number of vertices of $BFP(A)$ is $O(n)$, and that $BFP(A)$ consists of $O(n)$ simple arcs, each of which is either a line segment (a portion of a wall or of a radius emerging from C), an arc of E , or a portion of some *conchoid of Nicomedes*, which is a fourth-degree algebraic curve obtained as the locus of the endpoint P of the segment A as its other endpoint Q moves along a straight wall while it keeps touching the point C (see [Lo],[SS1] and the appendix for more detail).

Figure 1 : A portion of $FP(A)$

It is easily verified that each ray emerging from C intersects $FP(A)$ either in an open line segment or in the empty set. Using this property, we can apply to $FP(A)$ the following "angular" decomposition. For each vertex $v \in BFP(A)$ (other than C) draw a ray L_v which emerges from C , passes

through v and yields the line segment $FP(A) \cap L_v$. The set of these rays yields an angular decomposition $FP(A) = \bigcup_{i=1}^m c_i$ into $m = O(n)$ cells, so that each such cell c_i is bounded by two radial segments (contained in two adjacent delimiting rays, and one or both of which may degenerate to a single point), a "ceiling" which is either a line segment (along a wall) or a circular arc (of E), and a "floor" which is either an arc of a conchoid of Nicomedes (generated by a wall touched by Q), or the single point C (see Fig. 1). (For convenience, we will assume that each c_i contains its two bounding radial segments.)

In the sequel we will make use of the notion of the *radial shift* map $r_A: D - \{C\} \rightarrow D - \{C\}$, introduced in [SiS], which is defined so that for each $X \in D - \{C\}$ its image $r_A(X)$ is the point occupied by Q when the arm $A = PQ$ is placed with P at X (and with C touching PQ). It is clear (see also [SiS]) that r_A maps a segment of a radius of D onto a similar segment, that it is undefined at $X = C$, and that it maps a line segment L onto an arc of a conchoid of Nicomedes (defined by C and L), and vice versa.

In order to handle below the case of two Stanford arms, we next analyze the effect on $FP(A)$ of the insertion of a new wall edge W into the workspace D (such that W does not pass through C). W induces a modified smaller free configuration space, which we denote by $FP_W(A)$, and which can be written as $\bigcup_{i=1}^m FP_W(c_i)$, where $FP_W(c_i)$ is the restriction of $FP_W(A)$ to the cell c_i , $i = 1, \dots, m$. For each fixed cell c from the above angular decomposition, $FP_W(c)$ is the difference $c - FORB_W(c)$, where $FORB_W(c)$ is the set of *forbidden* (i.e. non-free) placements of A within $\bar{c}$ (the closure of c) at which A intersects W . $FORB_W(c)$ can be easily obtained as follows. First define the "shadow" of W with respect to C (and to D) as the set $SH(W) \subset \bar{D}$ consisting of points X for which there exists a point $Y \in W$ which belongs to the closed segment XC . Next, let $F_1 = SH(W) \cap \bar{c}$ and let $F_2 = SH(W) \cap r_A(\bar{c})$. Then $FORB_W(c) = F_1 \cup r_A(F_2)$ (see Fig. 2 for an example of $FORB_W(c)$).

It is easily seen that the boundary of $SH(W)$ for a line segment W consists of (the portion within D of) W , of the two radial extensions from its endpoints to the boundary E of D , and the (smaller) circular portion of E between the ends of these extensions.

We end this section by mentioning two additional properties of $FP_W(c)$.

- (P1) $FP_W(c)$ intersects the two radial edges of c in two respective sets, each of which is either a line segment or empty.
- (P2) Let γ be an arc from the boundary of $c \cup r_A(c)$. Then $SH(W) \cap \gamma$ is either a subarc of γ , or a single point of tangency between γ and $SH(W)$, or contains a finite number of intersection points, each of which is a transversal crossing of γ into $SH(W)$. This claim actually states that if γ is tangent to $SH(W)$ then they have a single point in common. Since $SH(W)$ is convex, the claim is obvious when γ is a line segment or a circular arc (of E). If γ is an arc of a conchoid of Nicomedes which is tangent to $SH(W)$, then the claim follows from the fact that a simple closed arc of such a conchoid bounds a convex

domain (see [Lo]), and that this domain is linearly separable from $SH(W)$ by their common tangent, as easily follows from definition.

c is the domain
bounded by K, L, M, N .

Figure 2 : An example of $FORB_W(c)$

4. The case of two independent planar Stanford arms

We next consider the case of the coordinated motion of two planar Stanford arms A_1 and A_2 modeled as two line segments P_1Q_1 and P_2Q_2 , which are free to move in the plane but are constrained to touch two fixed points C_1 and C_2 , respectively, not to intersect one another, and to avoid a collection of polygonal obstacles with a total of n corners, as above (see Fig. 3). The workspace D of the arms is the union of two discs D_1, D_2 centered at C_1, C_2 and having radii $|P_1Q_1|, |P_2Q_2|$ respectively. We denote the circle surrounding D_1 (resp. D_2) by E_1 (resp. E_2). Each arm A_i has two degrees of freedom which can be uniquely represented (under the restrictions made above) by the location of the arm's tip P_i , $i=1,2$. Each arm A_i induces its own "single-arm" free configuration space $FP(A_i)$, $i=1,2$, in which we assume the other arm A_{3-i} to be absent. These (2-D) spaces have the structure described in the preceding section.

Let us denote by $FP = FP(A_1, A_2)$ the (four-dimensional) free configuration space for both arms A_1, A_2 , and its boundary by $BFP(A_1, A_2)$. Each placement in $FP(A_1, A_2)$ can be represented as a point of $D_1 \times D_2 \subset \mathbb{R}^4$ (more precisely of $(D_1 - \{C_1\}) \times (D_2 - \{C_2\})$). To simplify the presentation of our algorithm, we assume that the obstacles lie in *general position* (with respect to the two arms) which means:

- (i) There is no collinear triplet $\{X_1, X_2, X_3\}$, where each X_i is either a vertex of $BFP(A_1) \cup BFP(A_2)$ or one of C_1, C_2 .

- (ii) For $i=1,2$, there is no collinear triplet $\{X_1, X_2, X_3\}$, with $X_1 \in \{C_1, C_2\}$, X_2 a vertex of $FP(A_i)$, and X_3 the image of a vertex of $FP(A_{3-i})$ under the radial shift $r_{A_{3-i}}$ (the reason for this somewhat peculiar condition is clarified later).

Figure 3: Two planar Stanford arms

$FP(A_1, A_2)$ can be described as follows. Let $c_1^1, \dots, c_{n_1}^1$ and $c_1^2, \dots, c_{n_2}^2$ be the cells which form the angular decomposition of $FP(A_1)$ and $FP(A_2)$, respectively; then a convenient decomposition of $FP(A_1, A_2)$ is

$$(*) \quad \bigcup_{i=1}^{n_1} \bigcup_{j=1}^{n_2} FP(A_1, A_2)|_{(c_i^1 \times c_j^2)}$$

where the subspace $FP(A_1, A_2)|_{(c_i^1 \times c_j^2)}$ (which we denote shortly as FP_{ij}) is the restriction of $FP(A_1, A_2)$ to the four-dimensional Cartesian product $c_i^1 \times c_j^2$, $1 \leq i \leq n_1$, $1 \leq j \leq n_2$. Following the general strategy outlined in the introduction, we next apply a cell-decomposition technique to each subspace FP_{ij} in the decomposition (*). Clearly each FP_{ij} has fixed complexity, because the coordinated motion of A_1 and A_2 within the two corresponding cells c_i^1 , c_j^2 is constrained by at most four walls, by two constraints on the range of allowed orientation of each arm, and by the constraint that the two arms avoid collision with one another. In what follows we provide more technical detail on how to obtain this (constant-size) decomposition in a convenient and economic manner; we note that a more global but closely related decomposition of FP has also been given in [FWY].

Let $BD(c^2)$ denote the circular sequence of arcs and vertices that constitute the boundary of $c^2 \equiv c_j^2$. $BD(r_{A_2}(c^2))$ denotes the corresponding sequence for the boundary of the radial-shift image r_{A_2} of c^2 . For each $X \in c^1 \equiv c_i^1$ let $A_1(X)$ denote the corresponding placement of A_1 . $A_1(X)$ can be

Figure 5 : Possible arcs for $FP_X(c^2)$

Let $BD^*(c^2) = BD(c^2) \cup BD(r_{A_2}(c^2))$. Then we have

Lemma 1: $X \in c^1$ is a critical point if and only if the line segment $A_1(X)$ is "tangent" to an element $e \in BD^*(c^2)$; more precisely, either an endpoint of $A_1(X)$ touches an arc of $BD^*(c^2)$, or $A_1(X)$ passes through a vertex of $BD^*(c^2)$, or $A_1(X)$ is tangent (in the standard sense) to an arc of $BD^*(c^2)$.

Proof: Let $X \in c^1$ be a point for which $A_1(X)$ is tangent to an element $e \in BD^*(c^2)$, in the above sense. If $e' = e \cap A_1(X)$ contains an infinite number of points, then e' must be a portion of a radius emerging from C_1 , and this situation is easily seen to contradict the assumption on general position. Hence we may assume that e' contains only a finite number of points. If e is an arc and $A_1(X)$ is tangent to e in the standard sense, then by property (P2) (at the end of Section 2), e' is a singleton, so that, if we move A_1 slightly in an appropriate direction to a new placement Y , we can clearly achieve a separation between e and $A_1(Y)$. The same property is easily seen to hold in the two other cases, in which either e is a vertex, or the "tangency" occurs at an endpoint of $A_1(X)$. It follows that there exist placements Y arbitrarily close to X at which the combinatorial representation of $FP_Y(c^2)$ is different from that of $FP_X(c^2)$, hence X is critical.

Conversely, suppose that $X \in c^1$ is a point for which $A_1(X)$ is not tangent to any element of $BD^*(c^2)$. Again by property (P2) and by the assumption on general position, each arc γ from $BD^*(c^2)$ either intersects $SH(A_1(X))$ transversally at a finite number of points, or the intersection between γ and $SH(A_1(X))$ is empty. Again it is easily checked that when Y varies sufficiently near X , $SH(A_1(Y))$ will continue to intersect γ in the same number of points, and be transversal to γ in each of them. This is easily seen to imply that the combinatorial representation of $FP_Y(c^2)$, for Y sufficiently near X , is

the same as that of $FP_X(c^2)$, hence X is non-critical. $\square$

Thus the set of critical points in c^1 is the union of finitely many *critical curves*, each containing placements X of A_1 at which $A_1(X)$ is tangent to some specific element from $BD^*(c^2)$ (in the sense of the last lemma). Each critical curve is of one of the following types:

(i) $\gamma(v)$:

The locus of points X at which $A_1(X)$ passes through some vertex $v \in BD^*(c^2)$, or is tangent (in the standard sense) at a point v to some arc β of $BD^*(c^2)$. $\gamma(v)$ is a subsegment of a radius which emerges from C_1 and passes through v . (It is easily checked that in the case of real tangency to some arc β , there are at most two such points v for each β .)

(ii) $\gamma(\lambda, P)$:

The locus of points X at which $A_1(X)$ touches some line segment $\lambda \in BD^*(c^2)$ at its endpoint P_1 . $\gamma(\lambda, P)$ is the segment λ itself (or more precisely, it is $\lambda \cap c^1$).

(iii) $\gamma(\alpha, P)$:

The locus of points X at which $A_1(X)$ touches some circular arc $\alpha \in BD^*(c^2)$ at its endpoint P_1 . $\gamma(\alpha, P)$ is the arc α itself (or more precisely, it is $\alpha \cap c^1$).

(iv) $\gamma(\kappa, P)$:

The locus of points X at which $A_1(X)$ touches some conchoidal arc $\kappa \in BD^*(c^2)$ at its endpoint P_1 . $\gamma(\kappa, P)$ is the arc κ itself (or more precisely, it is $\kappa \cap c^1$).

(v)-(vii) $\gamma(\lambda, Q), \gamma(\alpha, Q), \gamma(\kappa, Q)$:

These are the loci of points X at which $A_1(X)$ touches a line segment λ , a circular arc α , or a conchoid κ of $BD^*(c^2)$ at the endpoint Q_1 . These curves are simply the images (within c^1) of the corresponding arc λ, α , or κ under the radial shift r_{A_1} . They are also known as *conchoids*, having λ, α , or κ as their *base-curves* (see [Lo]). We will later analyze the algebraic equations defining these conchoids, and show that they have maximum degree 16.

The number of critical curves in c^1 is at most some fixed constant (independent of c^1 and c^2), and each of them is algebraic of degree at most 16. Hence they partition c^1 into at most some fixed constant number K of (relatively open) *non-critical regions* (i.e. connected components of the complement of the union of all critical curves). For each such region R , the combinatorial representation of $FP_X(c^2)$ clearly remains invariant as X varies within R . (This holds even for regions R that contain a portion of a radial segment e bounding c^1 ; that is, for $X \in e \cap R$ and for points $Y \in R$ approaching X from within the interior of c^1 , the representations of $FP_X(c^2)$ and of $FP_Y(c^2)$ are the same.) We denote this decomposition of c^1 by $c^1[c^2]$.

We can now obtain a cell decomposition of FP_{ij} as follows. Each cell in the decomposition has the form

$$\Lambda(R, L) \equiv \{[X, Y]: X \in R, Y \in \phi_X(L)\},$$

where R is a non-critical region of c^1 , L is a fixed circular labeling of a unique component of $FP_X(c^2)$, that we denote by $\phi_X(L)$. We represent FP_{ij} by a connectivity graph CG_{ij} , whose nodes are these cells, and whose edges represent adjacency of the cells within FP_{ij} . The "crossing rules" defining these edges are very similar to those developed in earlier papers on the cell decomposition technique (see [SS1],[SS3],[SS4]), and we therefore omit details of their analysis here. Roughly speaking, these rules are:

- (i) Each $\Lambda(R, L)$ is a connected cell.
- (ii) $\Lambda(R, L_1)$ and $\Lambda(R, L_2)$, for $L_1 \neq L_2$, are not adjacent.
- (iii) The cells $\Lambda(R', L)$ and $\Lambda(R'', L)$ (for the same label L) are adjacent, where R', R'' are non-critical regions adjacent along a common critical arc.
- (iv) The cells $\Lambda(R', L')$ and $\Lambda(R'', L'')$ are adjacent, where R', R'' are as in (iii), and L' (resp. L'') is any label associated with R' but not with R'' (resp. with R'' but not with R').

To analyze the crossing rules between different pieces FP_{ij} , we make use of the following

Remark: Each connected component of $FP(A_1, A_2)$ is a four-dimensional manifold. It is well-known that such a manifold cannot be disconnected by a submanifold whose dimension is less than three (see e.g. [HW] p. 48, and [SS1]), hence in developing the crossing rules between various portions of $FP(A_1, A_2)$, it suffices to consider only crossings which avoid some (finite number of) certain such submanifolds.

Consider now any single subspace FP_{ij} within the product $c_i^1 \times c_j^2$ from our basic decomposition $(*)$ of $FP(A_1, A_2)$. The two-dimensional cell c_i^1 (resp. c_j^2) has two radial boundary edges which may connect c_i^1 (resp. c_j^2) to its (possible) neighboring cells c_{i-1}^1 and c_{i+1}^1 (resp. c_{j-1}^2 and c_{j+1}^2). Moreover, crossing simultaneously from c_i^1 to a neighboring cell and from c_j^2 to a neighboring cell amounts to crossing a two-dimensional submanifold of the 4-D space FP . By the remark made above, the connectivity of FP does not change if we exclude from it (a finite number of) such submanifolds. Hence we can ignore such simultaneous crossings, and it suffices to analyze the crossing rules from the four-dimensional subspace FP_{ij} to its four neighboring subspaces $FP_{i-1,j}$, $FP_{i+1,j}$, $FP_{i,j-1}$, $FP_{i,j+1}$. To this end we distinguish between the following two cases: (1) crossing from FP_{ij} to the neighboring cells $FP_{i-1,j}$ or $FP_{i+1,j}$; (2) crossing from FP_{ij} to the neighboring cells $FP_{i,j-1}$ or $FP_{i,j+1}$.

Consider first the crossing rule to the subspace $FP_{i-1,j}$ ($FP_{i+1,j}$ is handled symmetrically). Let R' and R'' be two (open) non-critical regions which belong to the critical maps $c_i^1[c_j^2]$ and $c_{i-1}^1[c_j^2]$, respectively. Four different cases can arise.

- (i) $R' \cap R'' \neq \emptyset$:

In this case, this intersection must be a subsegment of the radial segment common to the boundaries of both c_l^1 and c_{l-1}^1 . For any $X \in R' \cap R''$, the combinatorial representation of $FP_X(c^2)$ is equal to the representation of $FP_Y(c^2)$ for any $Y \in R' \cap R''$. It follows that there exists a direct crossing between a pair of 4-D cells of the form $\Lambda(R', L'), \Lambda(R'', L'')$ if and only if $L' = L''$.

- (ii) $R' \cap R'' = \emptyset$, but $\bar{R}'$ and $\bar{R}''$ intersect each other in a line segment e :

In this case, e is again a subsegment of the radial segment common to the two cells c_l^1, c_{l-1}^1 , only this time it is a critical curve (of $c_l^1[c_j^2]$ as well as of $c_{l-1}^1[c_j^2]$). It is easily checked that in this case the line containing e must also contain C_1 , some vertex of c_l^1 , and some vertex of $BD^*(c_j^2)$. Since a vertex of $BD^*(c_j^2)$ is either a vertex of c_j^2 or the image of such a vertex under the radial shift r_{A_2} , this situation contradicts our assumption on general position, so we may ignore it.

- (iii) $R' \cap R'' = \emptyset$, but $\bar{R}' \cap \bar{R}''$ contains a finite set of points:

Let $X \in \bar{R}' \cap \bar{R}''$. Since the set $\{X\} \times FP_X(c_j^2)$ is two-dimensional, the preceding remark implies that we can also exclude such a crossing from our rules without affecting the connectivity of $FP(A_1, A_2)$.

- (iv) $\bar{R}' \cap \bar{R}'' = \emptyset$:

Clearly in this situation there does not exist a direct free crossing between any pair of cells projecting onto R' and R'' respectively.

Next we analyze the rule for crossing from $FP_{i,j}$ to $FP_{i,j-1}$, where $FP_{i,j+1}$ is handled symmetrically. Let R' and R'' be two non-critical regions belonging to the critical maps $c_l^1[c_j^2]$ and $c_l^1[c_{j-1}^2]$, respectively. We have to consider the following three cases.

- (i) $R' \cap R'' \neq \emptyset$:

Suppose there is a crossing between $\Lambda(R', L')$ and $\Lambda(R'', L'')$ for some pair of labels L', L'' . In this crossing A_2 crosses from c_j^2 to c_{j-1}^2 , so its tip must cross the radial edge e common to the boundaries of these two cells, and thus both labels L', L'' must include that radial edge. We claim that this is also a sufficient condition for the crossing in question to be possible. Indeed in this case, for any point $X \in R' \cap R''$, the two cells $\phi_X(L'), \phi_X(L'')$ are bounded by portions of the common radial edge e , and since $FP_X(c_j^2)$ and $FP_X(c_{j-1}^2)$ both intersect e in the same connected subsegment (by property (P1) mentioned in Section 3), it follows that one can cross between $\Lambda(R', L')$ and $\Lambda(R'', L'')$ by simply moving A_1 to any point X in $R' \cap R''$, and then move A_2 through the segment $FP_X(c_j^2) \cap e = FP_X(c_{j-1}^2) \cap e$.

- (ii) $R' \cap R'' = \emptyset$, but $\bar{R}' \cap \bar{R}'' \neq \emptyset$:

In this case, the placements where crossing can take place form a 2-D submanifold of $FP(A_1, A_2)$ (because A_1 then lies in the 1-D set $\bar{R}' \cap \bar{R}''$ and A_2 lies on the common radial edge e defined above). Hence by the preceding remark we may exclude such crossings from consideration.

(iii) $\bar{R}' \cap \bar{R}'' = \emptyset$:

Clearly in this situation there does not exist a direct free passage between any pair of cells $\Lambda(R', L'), \Lambda(R'', L'')$.

In summary, we have obtained

Theorem 1: (a) The combinatorial complexity of $FP(A_1, A_2)$ is $O(n^2)$.

(b) A combinatorial representation of $FP(A_1, A_2)$ in terms of a discrete connectivity graph $CG(A_1, A_2)$ can be constructed in $O(n^2)$ time and space, using the above cell decomposition technique.

(c) Any motion planning instance for A_1, A_2 can then be solved in time $O(n^2)$ by path searching in $CG(A_1, A_2)$.

Remarks: (1) Concerning the issue of lower bounds for our motion coordination problem, we note that one can not improve the upper bound in Theorem 1 by considering only a single connected component of $FP(A_1, A_2)$ (as discussed in Section 2), because (as illustrated in Fig. 6) a connected component of $FP(A_1, A_2)$ can have $\Omega(n^2)$ vertices.

Figure 6: A component of $FP(A_1, A_2)$ having $\Omega(n^2)$ complexity

Furthermore, in this example each cell c_i^1 in $FP(A_1)$, $1 \leq i \leq n_1$, intersects each cell c_j^2 in $FP(A_2)$, $1 \leq j \leq n_2$ (where both n_1 and n_2 are $\Theta(n)$); hence the number of cells $c_i^1 \times c_j^2$, $1 \leq i \leq n_1$, $1 \leq j \leq n_2$, for which $c_i^1 \cap c_j^2 \neq \emptyset$ (i.e. in which the two arms A_1 and A_2 can interfere with one another within $c_i^1 \times c_j^2$) is $\Omega(n^2)$. However, this example does not provide any lower bound for an algorithm to coordinate the free motion of two planar Stanford arms. We note that it is an open problem whether one can coordinate the free motion of two independent robot systems B_1 and B_2 in time which is substantially less than $O(n_1 n_2)$, where n_1 and n_2 are the times needed to separately plan the free motion of B_1 and of B_2 , respectively (each moving in the presence of

static obstacles only).

(2) Fortune, Wilfong and Yap [FWY] have also presented an $O(n^2 \log n)$ algorithm for coordinating the motion of two *dependent* planar Stanford arms, in which the two respective tips of the arms are constrained to remain in contact with each other as they move (namely, to be co-located). The free configuration space for this case is denoted by $FP(A_1 * A_2)$, and can be represented simply as the intersection $FP(A_1) \cap FP(A_2)$ (where, as above, both spaces are represented by the position of the common tip). The algorithm in [FWY] calculates this intersection by using a standard plane sweep, which takes time $O((n+p) \log n)$ where p is the actual number of intersections between the boundaries of $FP(A_1)$ and $FP(A_2)$ (p is $\Omega(n^2)$ in the worst case). We note here that by using the above angular decompositions of $FP(A_1)$ and $FP(A_2)$, we can compute $FP(A_1 * A_2)$ in time $O(n^2)$ only. Indeed, note that

$$FP(A_1) \cap FP(A_2) = \bigcup_{i=1}^{n_1} \bigcup_{j=1}^{n_2} (c_i^1 \cap c_j^2),$$

Figure 7: $FP(A_1 * A_2)$ has $\Omega(n^2)$ complexity

and for each pair of indices (i, j) , $1 \leq i \leq n_1$, $1 \leq j \leq n_2$, we can compute $c_i^1 \cap c_j^2$ (and glue it to its appropriate neighbors) in constant time. This shows that the overall complexity of $FP(A_1 * A_2)$ is $O(n^2)$, and that it can be computed in $O(n^2)$ time. Moreover, an easy example (illustrated in Fig. 7) shows that in the worst case the complexity of $FP(A_1 * A_2)$ can be $\Omega(n^2)$, thus the above algorithm is optimal in the worst case, if the entire space $FP(A_1 * A_2)$ is to be constructed.

However, we can actually do much better. Indeed, as has been observed e.g. in [PSS],[GSS] (see also the discussion in section 2), it suffices to calculate only the connected component C of $FP(A_1 * A_2)$ containing the initial

configuration Z of the two arms, since no other part of $FP(A_1 * A_2)$ can be reached from that configuration by a free motion. But C is simply the connected component of the intersection $FP(A_1) \cap FP(A_2)$ which contains Z . A "combination lemma" presented in [GSS] implies that the combinatorial complexity of C is linear in the combinatorial complexity of $FP(A_1)$, $FP(A_2)$, and is thus $O(n)$. Moreover, the "red-blue merge" procedure of [GSS] can be used to calculate C in time $O(n \log n)$. We thus obtain

Theorem 2: One can plan the coordinated motion of two planar Stanford arms, where the tips of the two arms are constrained to touch one another, in time $O(n \log n)$.

5. Coordinating the motion of two independent discs (or translating convex objects) in the plane

Next we consider the special case in which the two independently moving robots are two discs B_1 and B_2 having radii r_1 and r_2 , respectively, which are free to move in a 2-D polygonal region as in the previous sections. As noted in the introduction, this problem has been studied in [SS3] who obtain an $O(n^3)$ algorithm, using their cell decomposition technique, and in [Ya1] who gives an improved $O(n^2)$ algorithm, based on a certain retraction of FP onto its boundary BFP . Recently [RA] have also proposed a new $O(n^2)$ algorithm, but the details of their algorithm are somewhat unclear. Here we show that this problem can be handled in much the same way as the case of two Stanford arms, and obtain an $O(n^2)$ algorithm which, although does not improve the result of Yap [Ya1], puts this problem in the more general framework of coordinated motion planning that we have developed. Also our algorithm can be extended to handle coordinated translational motion of any pair of independent fairly arbitrary convex objects in the plane, provided the complexity of each of them is small and fixed.

In the case of discs, the space $FP(B_1)$ (and similarly $FP(B_2)$) is obtained by "growing" each obstacle by the radius of the disc B_1 , and forming the complement of the union of the expanded obstacles. It follows e.g. from the analysis of [KLPS] that the boundary of each of these spaces is composed of $O(n)$ line segments and circular arcs, and that these two spaces can be calculated in time $O(n \log^2 n)$ (as in [KLPS]), or by an improved $O(n \log n)$ algorithm based on Voronoi diagrams (see [OY] and [Ya2]). It is easily seen that $FP(B_1)$ (resp. $FP(B_2)$) can be "vertically decomposed" into $n_1 = O(n)$ (resp. $n_2 = O(n)$) simple cells $c_1^1, \dots, c_{n_1}^1$ (resp. $c_1^2, \dots, c_{n_2}^2$) such that the boundary of each cell c_i^1 (resp. c_j^2) consists of a "floor" and a "ceiling" (which are line segments or circular arcs of radius r_1 (resp. r_2) bounding some expanded obstacles) and of two straight vertical edges (one of which may degenerate to a single point) which connect c_i^1 (resp. c_j^2) to its neighboring cells (see Fig. 8). Assuming general position of the obstacles, it is easily seen that each of these cells can have at most four neighboring cells.

Figure 8: The vertical decomposition of $FP(B_1)$

As in Section 4 we can decompose $FP(B_1, B_2)$ as

$$(**) \quad \bigcup_{i=1}^{n_1} \bigcup_{j=1}^{n_2} FP(B_1, B_2)|_{(c_i^1 \times c_j^2)} .$$

In each subspace $FP_{ij} \equiv FP(B_1, B_2)|_{(c_i^1 \times c_j^2)}$, $1 \leq i \leq n_1$, $1 \leq j \leq n_2$, we can apply a cell decomposition technique similar to that of Section 4, where (in the notations introduced above) for each $X \in c_i^1$ the set $FP_X(c_j^2)$ is the set difference of c_j^2 and a closed circle centered at X and having radius $r_1 + r_2$. The critical curves that arise in c_i^1 are simpler than those obtained in the case of Stanford arms -- they are all circular arcs and straight segments (see [SS3] for more detail). Thus one can obtain (in constant time) an appropriate discrete connectivity graph CG_{ij} which represents FP_{ij} . The gluing procedure of the $n_1 n_2$ graphs CG_{ij} into an overall connectivity graph $CG(B_1, B_2)$ can be done in a similar manner to the gluing procedure in Section 4, in constant time per each pair of adjacent cells. We omit the technical details of developing the relevant crossing rules (which are similar in spirit to those obtained in Section 4).

We also note that the analysis presented above can be easily generalized to the case in which B_1, B_2 are convex objects translating in the plane, provided that the complexity of B_1 and of B_2 is small and fixed (e.g. they are both convex polygons with a small number of edges). The results of [KLPS] imply that the complexity of $FP(B_1), FP(B_2)$ is still $O(n)$, and all the other steps of the algorithm can be similarly generalized. (Note that we require that B_1 and B_2 have small fixed complexity to ensure that each of the spaces FP_{ij} involves only $O(1)$ geometric constraints and thus has constant complexity and can be calculated in constant time.) Omitting details of the fairly straightforward extension of the algorithm, we conclude

Theorem 3: Coordinated motion planning for two independent discs (or translating convex objects of small fixed complexity) in a 2-D polygonal region bounded by n edges, can be accomplished in $O(n^2)$ time.

6. Conclusion

The new approach to coordinated motion planning, as presented in this paper, can be applied to many other cases of two independently moving robot systems B_1 and B_2 , yielding a simple and general algorithm for calculating $FP(B_1, B_2)$, whose complexity is proportional to the product of the combinatorial complexities of $FP(B_1)$ and of $FP(B_2)$ (or of their relevant connected components). The main condition which is required for our technique to apply, is the capability to decompose $FP(B_1)$ and $FP(B_2)$ into collections of cells, each having a simple shape with $O(1)$ complexity, so that, within each Cartesian product of two such cells, $FP(B_1, B_2)$ has only $O(1)$ combinatorial complexity. If $FP(B_1), FP(B_2)$ are algebraic, then such decompositions generally exist (e.g. the Collins decomposition [SS2] has the required property). However, the number of cells needed to form such a decomposition (let us call it a "triangulation") may generally be much larger than the "actual" combinatorial complexity of FP (as measured e.g. by the complexity of its boundary). For 2-D spaces, Euler's formula generally makes such triangulations of small size easy to obtain, but it appears to be an open problem whether a three-dimensional algebraic variety can be triangulated into a number of cells proportional (or closely related) to its combinatorial complexity.

We also note that even when FP can be triangulated into a small number of simple cells, it might be the case that the complexity of FP_{ij} (within a single product cell $c_i^1 \times c_j^2$) is not constant. This may be the case when the moving systems B_1, B_2 have non-constant complexity. For example, if B_1 and B_2 are two simple polygons having k_1 and k_2 vertices, respectively, then the complexity of FP_{ij} will generally be a polynomial (of rather high degree) in k_1 and k_2 , so our approach does not yield as efficient an algorithm as those presented above. The difficulty can sometimes be overcome using ad-hoc techniques. For example, in the case of two polygons, we could have decomposed each of the polygons into $O(k_1)$ and $O(k_2)$ triangles respectively, solve the problem for each pair of such triangles (one from each polygon), and finally intersect the resulting spaces (as in the case of dependent Stanford arms) to obtain the desired FP .

As already mentioned earlier, we note that our approach is quite flexible in the manner in which FP_{ij} is to be calculated within each product cell $c_i^1 \times c_j^2$. One can use the Collins cell decomposition technique [SS2], the algebraic roadmap technique of Canny [Ca], or any other specific convenient technique which yields reasonable discrete structures that represent each FP_{ij} . The gluing of these structures to one another can also be accomplished by any convenient method (e.g. by calculating the restriction of FP to $c_i^1 \times (c_j^2 \cup c_{j \pm 1}^2)$ or $(c_i^1 \cup c_{i \pm 1}^1) \times c_j^2$ (in constant time) and deducing from these structures the adjacency relationships across cells). However, our technique can be applied only if each of the "underlying" spaces $FP(B_1), FP(B_2)$ can be decomposed

into a small number of simple cells, so only cell decomposition techniques appear to be adequate for the initial calculation of these spaces.

As an example where all these conditions are satisfied, we can extend our technique to obtain an $O(n^4)$ coordinated motion planning algorithm for two independent line segments ("rods") translating and rotating in a 2-D polygonal space with n corners, because each of the spaces $FP(B_i)$ can be represented by a cell decomposition of complexity $O(n^2)$ (by the analysis in [LS]), and the complexity of each rod is small and fixed.

Similarly, a recent algorithm in [AS] gives a cell decomposition of a single component of the free space of an arbitrary polyhedron (of $O(1)$ complexity) translating in 3-space amidst polyhedral obstacles having n faces altogether. This decomposition has at most $O(n^{7/3+\epsilon})$ cells, for any $\epsilon > 0$. Hence, using this algorithm, we can obtain an $O(n^{14/3+\epsilon})$ algorithm for coordinated motion planning of two independent translating polyhedra.

In contrast, several efficient algorithms have been recently obtained for various moving systems, which produce different representations of the free space FP , and do not seem to yield comparably efficient cell decompositions. A typical example is the algorithm in [KS] for a (translating and rotating) convex polygon in a 2-D polygonal space. Such algorithms do not immediately generalize under our technique to efficient coordinated motion planning algorithms for pairs of such systems.

Another important practical observation is that in certain cases it is possible for many subspaces FP_{ij} to coincide with the corresponding product cells $c_i^1 \times c_j^2$. For example, in the case of two independent discs, if the radii of the discs are much smaller than the dimensions of the obstacles, any subset FP_{ij} , for which the two discs lie in different (and relatively distant) portions of the workspace, will coincide with $c_i^1 \times c_j^2$. This observation implies that there is no need to calculate these FP_{ij} 's, or even to store them explicitly. Although we have not explored this observation in full, we regard it as a potentially significant source for further optimization of our algorithm. Of course, if the two systems tend to be generally near one another, so that interference between them is likely to occur in many product cells, then this optimization will fail to be significant (see Fig. 6 for example).

Next we make the following general observation concerning coordinated motion planning. If the two moving systems have respectively k_1, k_2 degrees of freedom, then by the general recent results of Canny [Ca], coordinated motion planning for this pair of systems (which has $k_1 + k_2$ degrees of freedom) can be accomplished in time close to $O(n^{k_1+k_2})$ (again, ignoring the algebraic complexity of the constraints). However, whenever the motion planning problem for each of the systems separately can be solved in better than $O(n^{k_1})$ or $O(n^{k_2})$ time (by a cell decomposition technique), our technique yields an improved algorithm for the coordinated problem (e.g. in the case of two independent rods mentioned above, our $O(n^4)$ algorithm improves the $O(n^6 \log n)$ algorithm yielded by Canny's procedure).

Finally, a natural generalization of our approach is to apply it to k independent robot systems $B_1, \dots, B_k$. It seems that our approach can

potentially yield a decomposition of $FP(B_1, \dots, B_k)$ into "local" subspaces which can be properly glued to one another to form the entire space, so that the total time and space needed to obtain this representation is $O(C(k) \prod_{i=1}^k n_i)$, where n_i is the combinatorial complexity of a cell decomposition of $FP(B_i)$, $1 \leq i \leq k$ (or of some specific connected component thereof), and $C(k)$ is an upper bound for the time needed to calculate $FP(B_1, \dots, B_k)|_{(c^1 \times \dots \times c^k)}$ (where each c^i is a cell in the decomposition of $FP(B_i)$, $1 \leq i \leq k$). In particular, if $B_1, \dots, B_k$ are k planar discs (or k planar Stanford arms), then our technique would yield an $O(C(k)n^k)$ algorithm for calculating $FP(B_1, \dots, B_k)$, thus reproducing, and generalizing, the results of [RA]. The constant $C(k)$ will in general be quite large. The recent results of [Ca] suggest that it might be only singly exponential in k .

REFERENCES

- [ASS] P. Agarwal, M. Sharir and P. Shor, Sharp upper and lower bounds for the length of general Davenport Schinzel sequences, Tech. Rept. 332, Comp. Sci. Dept., Courant Institute, November 1987.
- [AS] B. Aronov and M. Sharir, Triangles in space, or: Building (and analyzing) castles in the sky, *Proc. 4th ACM Symp. on Computational Geometry*, 1988.
- [Ca] J. Canny, The complexity of robot motion planning, Ph.D. Dissertation, Comp. Sci. Dept., M.I.T., May 1987.
- [EL] M. Erdman and T. Lozano-Perez, On multiple moving objects, *Algorithmica* 2 (1987) pp. 477-521.
- [FWY] S. Fortune, G. Wilfong and C. Yap, Coordinated motion of two robot arms, *Proc. IEEE Conference on Robotics and Automation*, pp. 1216-1223, April 1986.
- [GSS] L.J. Guibas, M. Sharir and S. Sifrony, On the general motion planning problem with two degrees of freedom, *Proc. 4th ACM Symp. on Computational Geometry*, 1988.
- [HS] S. Hart and M. Sharir, Nonlinearity of Davenport Schinzel sequences and of generalized path compression schemes, *Combinatorica* 6 (1986), pp. 151-177.
- [HW] W. Hurewicz and H. Wallman, *Dimension Theory*, Princeton University Press, 1948.
- [KLPS] K. Kedem, R. Livne, J. Pach and M. Sharir, On the union of Jordan regions and collision-free translational motion amidst polygonal obstacles, *Discrete Comput. Geom.* 1 (1986), pp. 59-71.
- [KO] Y. Ke and J. O'Rourke, Moving a ladder in three dimension: Upper and lower bounds, *Proc. 3rd ACM Symp. on Computational Geometry*, pp. 136-146, 1987.
- [KS] K. Kedem and M. Sharir, An efficient motion planning algorithm for a convex rigid polygonal object in two-dimensional polygonal space, to appear in *Discrete Comput. Geom.*
- [Lo] E.H. Lockwood, *A Book of Curves*, Cambridge University Press, 1967.
- [LS] D. Leven and M. Sharir, An efficient and simple motion-planning algorithm for a ladder moving in 2-dimensional space amidst polygonal barriers, *J. Algorithms* 8 (1987), pp. 192-215.
- [OY] C. O'Dunlaing and C. Yap, A 'retraction' method for planning the motion of a disc, *J. Algorithms* 6 (1985) pp. 104-111.

- [OSY1] C. O'Dunlaing, M. Sharir and C. Yap, Generalized Voronoi diagrams for a ladder: I. Topological considerations, *Comm. Pure Appl. Math.* 39 (1986), pp. 423-483.
- [OSY2] C. O'Dunlaing, M. Sharir and C. Yap, Generalized Voronoi diagrams for a ladder: II. Efficient construction of the diagram, *Algorithmica* 2 (1987), pp. 27-59.
- [PSS] R. Pollack, M. Sharir and S. Sifrony, Separating two simple polygons by a sequence of translations, *Discrete Comput. Geom.* 3 (1988) pp. 123-136.
- [RA] G. Ramanathan and V.S. Alagar, Algorithmic motion planning in robotics: Coordinated motion of several discs amidst polygonal obstacles, *Proc. IEEE Conference on Robotics and Automation*, 1985, pp. 514-522.
- [SS1] J.T. Schwartz and M. Sharir, On the piano movers' problem: I. The case of a rigid polygonal body moving amidst polygonal barriers, *Comm. Pure Appl. Math.* 36 (1983) pp. 345-398.
- [SS2] J.T. Schwartz and M. Sharir, On the piano movers' problem: II. General techniques for computing topological properties of real algebraic manifolds, *Adv. Appl. Math.* 4 (1983) pp. 298-351.
- [SS3] J.T. Schwartz and M. Sharir, On the piano movers' problem: III. Coordinating the motion of several independent bodies: The special case of circular bodies moving amidst polygonal barriers, *Int. J. Robotics Research* 2 (1983) (3) pp. 46-75.
- [SS4] J.T. Schwartz and M. Sharir, Efficient motion planning algorithms in environments of bounded local complexity, Tech. Rept. 164, Comp. Sci. Dept., Courant Institute, June 1985.
- [SiS] S. Sifrony and M. Sharir, A new efficient motion-planning algorithm for a rod in 2-D polygonal space, *Algorithmica* 2 (1987) pp. 367-402.
- [Ya1] C.K. Yap, Coordinating the motion of several discs, Tech. Rept. 105, Comp. Sci. Dept., Courant Institute, February 1984.
- [Ya2] C.K. Yap, An $O(n \log n)$ algorithm for the Voronoi diagram of a set of simple curve segments, *Discrete Comput. Geom.* 2 (1987) pp. 365-393.

Appendix A : Analysis of conchoids and calculation of critical points

In this appendix we present a few technical details pertaining to the algebraic structure of the critical curves arising in the case of two Stanford arms. These curves are line segments, circular arcs or conchoidal arcs, and we consider here only the conchoids. Let γ be some simple (algebraic) curve in the plane, and let $B=PQ$ be a line segment which moves in the plane such that the endpoint Q touches the curve γ and B is constrained to touch a fixed point C . Then as noted in Section 4, the other endpoint P of B moves along a conchoid $\kappa(\gamma, C)$ (having γ as a base-curve and C as a fixed-point). Without loss of generality assume that C is the origin of our coordinate system. Let $P=(x,y)$ and $Q=(x',y')$. If d denotes the length of PQ , then it is easily checked that (x,y) and (x',y') are related by (see also Fig. 9)

Figure 9: A conchoid

$$x' = x \left(1 - \frac{d}{\sqrt{x^2 + y^2}}\right)$$

$$y' = y \left(1 - \frac{d}{\sqrt{x^2 + y^2}}\right)$$

If γ is the straight line $y' = ax' + b$, then $\kappa(\gamma, C)$ is a conchoid of Nicomedes (see also [SS1]) whose equation is thus

$$(x^2 + y^2)(y - ax - b)^2 = d^2(y - ax)^2.$$

It is easily verified that if γ is a quadric, then the equation defining the conchoid $\kappa(\gamma, C)$ is algebraic of degree 8; and in general if γ is algebraic of degree k , then $\kappa(\gamma, C)$ is algebraic of degree (at most) $4k$. In particular, if γ is a conchoid of Nicomedes (i.e. γ is of the fourth degree), then $\kappa(\gamma, C)$ is algebraic of degree 16.

Thus the degree of any of our critical curves is at most 16. Since the computation of the decomposition of c_l^1 into non-critical regions involves intersections of pairs of critical curves, we may face (repeatedly) the task of intersecting pairs of algebraic curves of fairly large degree, which can be quite cumbersome. Nevertheless, we have

Lemma A.1: Any intersection between a pair of critical curves in c_l^1 can be obtained by solving a pair of algebraic equations one of which is of degree at most 4.

Proof: If the degree of both curves to be intersected is greater than 4, then each of them must be of the form $\gamma(\alpha, Q)$ or $\gamma(\kappa, Q)$. But then we can express the desired intersection in terms of the position of the endpoint Q_1 , which is plainly obtained by solving a pair of equations, each of degree at

most 4. $\square$

The proof of Lemma A.1 is easily seen to imply that the most difficult calculation that we may encounter is of the intersection between a conchoid of Nicomedes and an algebraic curve $P(x,y)=0$ of degree at most 16. Actually, using a more careful analysis, we can show that the intersection of two critical curves can always be obtained by solving a pair of equations, such that either both are of degree at most four, or one of them is quadratic (and the other is of degree at most 16).

NYU COMPSCI TR-360 c.1
Sharir, Micha
Coordinated motion planning
for two independent robots

NYU COMPSCI TR-360 c.1 -
Sharir, Micha
Coordinated motion planning
for two independent robots

This book may be kept JUN 10 1988

FOURTEEN DAYS

A fine will be charged for each day the book is kept overtime.